ARMY RESEARCH LABORATORY

STATBIC—A Method for Inclusion of Fractal Statistics in Obscurant Transport Models

by Sean G. O'Brien and Donald W. Hoock

ARL-TR-1375 April 1998

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory
White Sands Missile Range, NM 88002-5501

ARL-TR-1375 **April 1998**

STATBIC—A Method for Inclusion of Fractal Statistics in Obscurant Transport Models

Sean G. O'Brien
Physical Science Laboratory, New Mexico State University

Donald W. Hoock
Information Science and Technology Directorate, ARL

Abstract

A generic texture routine was developed for upgrading smooth obscurant cloud models by introduction of time- and space-dependent fluctuations in line-of-sight (LOS) propagation and image generation. The routine runs separately from or in conjunction with other obscuration models that predict electro-optical (EO) propagation for mean or average aerosol concentration contributions in the obscurant cloud.

Atmospheric turbulence and eddy structures are the underlying sources of concentration fluctuations. Eddy structures can possess certain well-defined and statistically predictable correlations between larger and smaller eddies that are linked to the steady breakup of larger eddies and the cascade of energy to smaller eddy scales. The resulting effects of these correlations in concentration are simulated by prototype algorithms that provide either two-dimensional propagation overlays for image modification or three-dimensional volume fluctuations. Path-integrated concentration, LOS propagation fluctuations, and realistic cloud imaging are then simulated by multiplication of pseudo-random fluctuation outputs with smooth, ensemble-averaged outputs from more traditional, simple obscuration models.

Two platform-specific implementations of the generic texture routine were developed for IBM PC and Silicon Graphics, Inc. (SGI), display hardware. Examples of graphical output for both the two and three-dimensional algorithms are shown.

Contents

Figures

1. Introduction

The codes in use today for modeling screening obscurant transport and diffusion are frequently designed for rapid execution in war games. Because of this limitation, the transport and diffusion equations in these models are simplified. In modeling the down-wind transport and diffusive growth of an aerosol cloud under the influence of nonuniform wind and temperature fields, the aerosol cloud is represented by a Gaussian puff approximation (or some similar structural primitive). Consequently, at any given time after release of an obscurant, the spatial concentration distribution predicted by such models is normally analytic or "smooth." In reality, the mechanical turbulence that is caused by the obscurant dissemination process (or that is present in the natural atmosphere) will produce concentration fluctuations in the obscurant plume. A real plume will thus show thick knots and holes where significant deviations from the mean concentration field are present. It is thus likely that the smooth concentration distribution of obscurant in these models does not properly represent cumulative engagement probabilities for opposing units composed of multiple combat entities [1].

For example, a combatant viewing a given spatial distribution of targets through an obscurant field that is fluctuating in both time and space can exploit transient holes in the obscurant screen, hitting more targets than would be estimated on the basis of a smooth concentration field. If more than one such combatant is viewing the same target field, a given target may be exposed to more than one opponent and thus have an even lower probability of survival relative to the average concentration estimate.

One means of improving the statistical accuracy of transport models is to include stochastic processes on a microphysical scale. For example, large eddy simulation models treat the hydrodynamics of turbulent flow by numerically solving the Navier-Stokes equations, and provide a physically accurate method for determining the fluctuating concentration field. Unfortunately, such models involve large numbers of calculations, and are thus too slow to employ in wargaming simulations.

An alternative and faster approach to simulation of turbulent transport is first to use the simplified-transport methodology to establish the mean concentration field, and then to apply multiplicative factors to the mean that are determined by the Kolmogorov statistics of steady-state turbulence in the boundary layer. This approach is used for the three-dimensional (3D) variant of the Statistical Texturing Application to Battlefield-Induced Clouds (STATBIC) algorithm, called STATBIC-3D [2]. For situations where a 2D image of the obscurant cloud embedded in a scene is to be generated, STATBIC-2D has been developed to apply a Kolmogorov multiplicative factor for the column density of obscurant (also known as the concentration length) over a given line of sight (LOS) [2].

2. Theory

The objective of the STATBIC algorithms is to generate pseudo-random 4D realizations of LOS obscuration that are consistent with the statistics of spatial and temporal correlations within the turbulent boundary layer of the atmosphere. The homogeneous Kolmogorov turbulence field that is assumed to be present in the boundary-layer region obeys a –5/3 power-law roll-off in spectral power density. The spatial frequency region over which the power law pertains is known as the inertial subrange. Higher spatial frequencies (smaller eddies) above the inertial subrange are subject to viscous friction damping and thus depart from the simple power-law cascade behavior. Below the inertial subrange, in the spatial frequency region known as the input range, the spectrum of the large eddies is driven by boundary-layer depth, wind-field structure, and temperature-field structure, resulting in large departures from Kolmogorov behavior.

In the inertial subrange (where STATBIC is applicable), the power-law characteristic implies that the turbulence-modulated concentration $C(x,t)$ may be modeled as having fractal statistics and scaling properties [3]. If one wishes to find the mean-square fluctuation for time $t + \Delta t$ after initial time t (for a fixed point in space) or position $x + \Delta x$ separated from position x (at a fixed time), the fractal assumption yields the relations

$$\left\langle \left| C(t + \Delta t) - C(t) \right|^2 \right\rangle = (\Delta t)^{2H} \sigma_t^2 \ , \tag{1}$$

and

$$\left\langle \left| C(x + \Delta x) - C(x) \right|^2 \right\rangle = (\Delta x)^{2H} \sigma_x^2 \ , \tag{2}$$

where σ_t^2 and σ_x^2 are the variances at unit increments of time and position, respectively. The Hurst parameter $H = E - D$ is the difference between the Euclidean dimension E in which the fluctuations are embedded and the fractal dimension D of the fluctuation process. For Kolmogorov turbulence, H assumes the value of $1/3$.

We can use the above relations to examine the relationship between the Hurst parameter and the sense of correlation between concentrations at different times or positions. Suppose, for example, that the time interval Δt is built up as the sum of equal time steps δt, and that the fluctuation is simulated by addition of random numbers to an accumulator at each time step. If one wishes to mimic Brownian fluctuations with zero correlation, one might select random numbers from a Gaussian distribution with mean zero and standard deviation σ_t that are independent at each time step. For the Brownian process, a well-known result is that the mean-square fluctuation after n samples (or time interval $\Delta t = n\delta t$) is simply $n\sigma_t^2$. The mean-square fluctuations given by equations (1) and (2) increase linearly with n and imply that H has a value of $1/2$ in this instance. When H is larger than $1/2$, the mean-square fluctuation increases more rapidly than n with each step, an indication that correlation is positive between

successive random numbers in the process. Negative correlation holds when H is less than 1/2, as in Kolmogorov turbulence. In this latter case, there is a better than even chance that a fluctuation at a given time step will be in a direction opposite to what it was at the previous time step.

With the statistics of the Kolmogorov fractal process established for a given turbulent eddy size, the next step in the simulation of the process is to extend these results to an ensemble of eddy sizes. The self-affinity property of a concentration field $C(x,t)$, which obeys fractal statistics, means that the fluctuation difference functions

$$C(t + \Delta t) - C(t) \tag{3}$$

and

$$r^{-H}[C(t + r \cdot \Delta t) - C(t)] \tag{4}$$

should be statistically equivalent for any scale factor r, with corresponding relations for variations in position x. Addition of such scaled fluctuations (with $H = 1/3$) over a representative set of inertial subrange eddy sizes should produce a fractal field with Kolmogorov statistical properties.

An appropriate set of such summed basis functions is the Fourier series. Each term in the series represents a different eddy size, with amplitudes that are proportional to the scaling factor, and phase shifts $\varphi(x,t)$ that represent position in space and time. These phase shifts may be randomly varied to produce temporal variations and relative position shifts for the eddies. As implemented in STATBIC, random phase shifts in φ define the relative positions for different-sized eddies at a given time, with the distribution at subsequent times being simulated by time-correlated random shifts in φ. The phase decorrelation operation is performed in the spatial frequency domain (k-space) at each time step, so that only the inverse Fourier transform is required to produce the fluctuation distribution. The degree of correlation $R(k_j,t)$ at a given eddy spatial frequency k_j depends upon the mean eddy lifetime t_j, with an assumed exponential decay [2]:

$$t_j = \left(\frac{k_0}{k_j}\right)^{2/3} \frac{L_0}{u} \tag{5}$$

and

$$R\left(k_j, t\right) = e^{-t/t_j} \ , \tag{6}$$

where L_0 is the outer scale of turbulence, u is the horizontal wind speed, and $k_0 = 2\pi/L_0$. The phase $\varphi(k_j,t)$ is changed over time step Δt by the addition of a new independent random phase φ_{rv} (ranging from 0 to 2π) to the previous phase φ_j^{old}, producing a random phase φ_j^{new} that tends to $\varphi_j^{\text{old}} + \varphi_{rv}$ for Δt much larger than t_j:

$$\varphi_j^{\text{new}} = \varphi_j^{\text{old}} + \varphi_{rv}\,[1 - R_j(\Delta t)] \ . \tag{7}$$

With the composite phase $\varphi_{ijk} = \varphi_j^{new}$, the fluctuation f_C for concentration C at location (x,y,z) is then obtained by the 3D inverse Fourier transform

$$f_C(x,y,z) = \sum_{i=-N_x/2}^{N_x/2} \sum_{j=-N_y/2}^{N_y/2} \sum_{k=-N_z/2}^{N_z/2} \left[\frac{k_0}{\sqrt{k_0^2 + k_i^2 + k_j^2 + k_k^2}} \right]^{H+\frac{3}{2}} \times e^{i\left(k_i x + k_j y + k_k z + \varphi_{ijk}\right)} \tag{8}$$

with $k_i = 2\pi i/L_x$, $k_j = 2\pi j/L_y$, $k_k = 2\pi k/L_z$. L_x, L_y, and L_z are the largest scales along the x, y, and z axes, and N_x, N_y, and N_z are the number of volume elements along the x, y, and z axes, respectively. This fluctuation may then be applied to the mean concentration field $C_m(x,y,z)$ to produce the fluctuating concentration $C(x,y,z)$:

$$C(x,y,z) = C_m(x,y,z)\, f_C(x,y,z) \ . \tag{9}$$

This is the fundamental result output by the STATBIC-3D algorithm at each scenario time.

The 2D output of STATBIC-2D requires statistics on the LOS path integrated C_L rather than the concentration C. C_L is the path column density defined by the relation

$$C_L = \int_0^L C(L')\, dL' \ , \tag{10}$$

for a path of length L. Individual points in the STATBIC-2D image are parallel projections in which C_L fluctuations are functions of transverse point separation and path length. The integration of concentration over an LOS means that for two parallel LOSs of length L separated by a distance ρ, the C_L structure function may be expressed as a path average [2]:

$$\begin{aligned} &\left\langle \left| f_{CL}(L,\rho) - f_{CL}(L,0) \right|^2 \right\rangle \\ &\quad = \left(\frac{C_f}{L} \right)^2 \int_0^L ds \int_0^L ds' \left(\left[(s-s')^2 + \rho^2 \right]^{1/3} - (s-s')^{2/3} \right) \\ &\quad = 3.82 \left(\frac{L}{L_0} \right)^{2/3} \sigma_C^2 \int_0^1 dw (1-w) \left(\left[w^2 + \left(\frac{\rho}{L} \right)^2 \right]^{1/3} - w^{2/3} \right) , \end{aligned} \tag{11}$$

where use has been made of equation (2) for the concentration fluctuations with $H = 1/3$. The fluctuation strength C_f that appears in equation (11) is estimated in terms of the local mean gradient in concentration and the turbulence outer scale L_0:

$$C_f^2 = 2.8 \left(L_0 \right)^{4/3} \left| \frac{\nabla \langle C \rangle}{\langle C \rangle} \right|^2 = 1.91\, L_0^{-2/3}\, \sigma_C^2 \ , \tag{12}$$

where σ_C is the dimensionless standard deviation for concentration fluctuations.

Assuming that the C_L fluctuations scale in a manner analogous to those for the concentration C in equation (2), it is apparent that the slope of the log of the equation (11) integral with respect to the log of (ρ/L) (the ratio of path separation to path length) is twice the value of the Hurst parameter H. Figure 1 shows the numerically integrated and differentiated results for H as a function of ρ/L.

The Hurst parameter approaches a value of 5/6 (0.83) for small separations at a given L, which implies that the C_L fluctuations are relatively smooth for small separations ρ. When the LOS separation becomes larger, the C_L fluctuations are larger and show less correlation. At very large separations, H approaches the Kolmogorov point value of 1/3. Note also that H has a value of about 0.5 when ρ/L is unity, indicating that the fluctuations are Brownian, with zero correlation. In essence, for a fixed LOS, the fractal dimension for C_L fluctuations varies with transverse scale. This implies that the Fourier series representation of the fluctuation multiplier f_{CL} (similar to that given for f_C by eq (8)) is not strictly applicable for C_L fluctuations. Basis functions have been developed that account for the variable-scale Hurst parameter, but the resulting code will need to be further optimized to produce near real-time performance. At present, the f_{CL} multiplier is represented by a fixed H 2D-inverse Fourier transform:

$$f_C(x,z) = \sum_{i=-N_x/2}^{N_x/2} \sum_{j=-N_y/2}^{N_y/2} \sum_{k=-N_z/2}^{N_z/2} \left[\frac{k_0}{\sqrt{k_0^2 + k_i^2 + k_j^2 + k_k^2}} \right]^{H+\frac{3}{2}} \times e^{i(k_i x + k_k z + \varphi_{ik})} e^{-1/2\, k_j^2 \sigma_y^2} \ , \tag{13}$$

where σ_y is the standard deviation width of the cloud plume (or length of the LOS through the cloud), and the composite random phase φ_{ik} is φ_i^{new}. The fractal 2D C_L distribution is then given by

$$C_L(x,z) = (C_L(x,z))_m f_{CL}(x,z) \ , \tag{14}$$

where $(C_L(x,z))_m$ is the mean (smooth) Gaussian C_L value in the (x,z) projection plane.

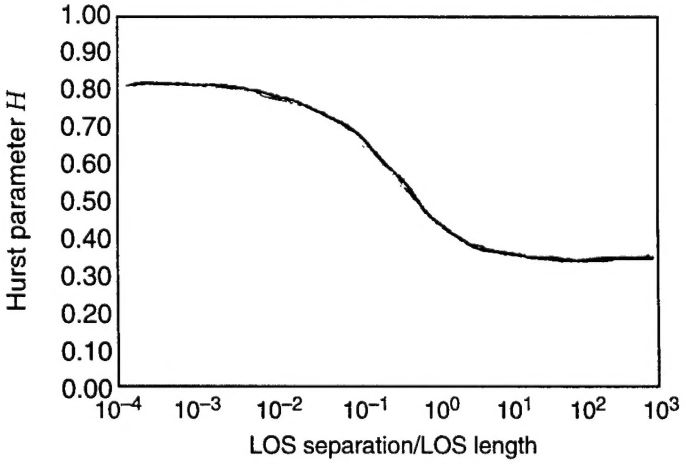

Figure 1. C_L Hurst parameter H as function of ratio of line of sight (LOS) separation r to LOS length L.

Unlike STATBIC-3D, which produces 3D concentration fields as its output, STATBIC-2D yields 2D maps of integrated path radiance L_p and transmission $T(R)$ over a path of length R. The radiance fields are calculated (in the present demonstration version of the algorithm) from fitted functional relationships between optical depth and limiting path radiance L_s for two example sun-cloud-observer relative geometries:

$$L_{\text{eye}} = 255\left(0.2 + \frac{1.273}{1 + 0.293\,\tau}\right),$$
$$L_{\text{back}} = 355\left(1.501 - \frac{1.2093}{1 + 0.0312\,\tau}\right), \tag{15}$$

where L_{eye} is the limiting path radiance for the sun at an elevation of 30° above the horizon in the observer's eyes, L_{back} is L_s for a sun 30° above the horizon at the observer's back, and τ is the optical depth. These results are fits to average values of L_s over the face of a cube of fog oil aerosol that faces the observer. They were produced by an early version of the BEAMS (Battlefield Emission and Multiple Scattering) model [4]. Figure 2 shows the behavior of the two fit curves with optical depth.

The optical depth τ (used to evaluate the fit function expressions in eq (15)) and the LOS transmission $T(R)$ are given by the relations

$$\tau(R) = \alpha_{\text{ext}} \int_0^R C(s)\,ds = \alpha_{\text{ext}}\, C_L\,,$$
$$T(R) = e^{-\tau}\,, \tag{16}$$

where α_{ext} is the mass extinction coefficient ($m^2\ g^{-1}$) for the obscurant. Under the assumption of uniform limiting path radiance over the LOS, the integrated path radiance $L_p(R)$ may be calculated and combined with background radiance $L(0)$ to provide the apparent total radiance $L(R)$ for the LOS:

$$L(R) = T(R)\,L(0) + L_p(R) \approx T(R)\,L(0) + [1 - T(R)]\,L_s\,. \tag{17}$$

Equation (17) may be viewed as a linear interpolation between a clear atmosphere background radiance $L(0)$ (for $T(R) = 1$) and a path radiance $L_p = L_s$ under totally obscured conditions ($T(R) = 0$).

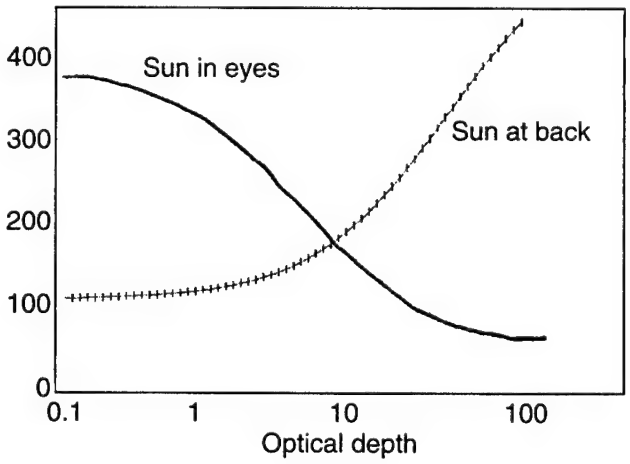

Figure 2. Fitted limiting path radiance curves for fog oil cloud, averaged over cube face that faces observer.

3. Implementation

3.1 General Comments

STATBIC-3D and STATBIC-2D are implemented through application of fractal multipliers f_C and f_{CL} to smooth individual Gaussian puffs or plumes, respectively. In principle, an arbitrary concentration distribution may be used in place of Gaussian distributions, but provision for user non-Gaussian input (aside from the trivial uniform distribution case) is not presently included in the STATBIC demonstration codes. It is left to the user either to modify the environmental parameter assignments within each code or to provide input/output (I/O) code that reads in environmental or obscurant plume parameters.

The fractal multiplier f_C for STATBIC-3D (eq (8)) is calculated with an inverse fast Fourier transform (FFT). We can enhance the efficiency of the FFT procedure if we note that the FFT of two real-valued functions (at two adjacent times) can be handled in a single transform by one of them being "packed" into the real part and the other into the imaginary part of the complex-valued argument of the transform [2,5]. STATBIC-3D employs the inverse procedure, in which complex random functions in frequency (k) space are inverse-transformed to two real functions at two adjacent times. Thus, at each frequency point (k_i, k_j, k_k), there are four statistically independent random phase coefficients chosen for time t_1 (H, J, L, and N) and four chosen for time t_2 (I, K, M, and O) for the 3D inverse transform. In order for the spatial output array to have its real part associated with time t_1 and imaginary part with time t_2, the following symmetries are imposed on the output function T [2]:

for $k_i > 0$, $k_j > 0$, and $k_k > 0$,
$$T = H + I + J + K + L + M + N + O\,;$$

for $k_i < 0$, $k_j > 0$, and $k_k > 0$,
$$T = H - I - J + K + L - M - N + O\,;$$

for $k_i > 0$, $k_j < 0$, and $k_k > 0$,
$$T = H + I - J - K - L - M + N + O\,;$$

for $k_i < 0$, $k_j < 0$, and $k_k > 0$,
$$T = H - I + J - K - L + M - N + O\,;$$

for $k_i > 0$, $k_j > 0$, and $k_k < 0$,
$$T = (H + I + J + K - L - M - N - O)^*\,;$$

for $k_i < 0$, $k_j > 0$, and $k_k < 0$,
$$T = (H - I - J + K - L + M + N - O)^*\,;$$

for $k_i > 0$, $k_j < 0$, and $k_k < 0$,
$$T = (H + I - J - K + L + M - N - O)^*\,;$$

for $k_i < 0$, $k_j < 0$, and $k_k < 0$,
$$T = (H - I + J - K + L - M + N - O)^*\,,$$

where the superscript asterisk in the last four relations implies complex conjugation. The output transform function T is then multiplied by the fractal filter function

$$F(k_i, k_j, k_k) = \left[\frac{k_0}{\sqrt{k_0^2 + k_i^2 + k_j^2 + k_k^2}} \right]^{H+\frac{3}{2}} \tag{18}$$

and is then transformed via an FFT implementation of equation (8),

$$f_C(x,y,z) = \sum_{i=-N_x/2}^{N_x/2} \sum_{j=-N_y/2}^{N_y/2} \sum_{k=-N_z/2}^{N_z/2} F(k_i, k_j, k_k)\, T(k_i, k_j, k_k) \times e^{i(k_i x + k_j y + k_k z)}, \tag{19}$$

to yield the fractal concentration multipliers f_C at t_1 and t_2. The sequence of transformations used in the equation (19) FFT is to transform first in z, then in y, and then in x (the reverse sequence to the forward transform). The f_C multipliers are then applied to the mean concentration according to the prescription of equation (9), and the resulting fractal concentration field is written to output files for the two scenario times.

STATBIC-2D follows a similar plan of implementation as that for STATBIC-3D, but in two dimensions. Instead of the four random coefficients required at each scenario time for the 3D case, only two are needed (D and F at time t_1 and E and G at time t_2). Symmetry considerations like those in the 3D case lead to the following constraints on the output function T (for an inverse transform from the (k_i, k_k) plane):

for $k_i > 0$ and $k_k > 0$,
$$T = D + E + F + G\,;$$

for $k_i < 0$ and $k_k > 0$,
$$T = D - E - F + G\,;$$

for $k_i > 0$ and $k_k < 0$,
$$T = (D + E - F - G)^*\,;$$

for $k_i < 0$ and $k_k < 0$,
$$T = (D - E + F - G)^*\,.$$

As in the 3D case, a 2D fractal filter $F(k_i, k_k)$ is calculated from the relation

$$F(k_i, k_k) = \left[\frac{k_0}{\sqrt{k_0^2 + k_i^2 + k_k^2}} \right]^{H+\frac{3}{2}} \tag{20}$$

and the output transform function is then inverse-transformed via an FFT implementation of equation (13):

$$f_{CL}(x,z) = \sum_{i=-N_x/2}^{N_x/2} \sum_{k=-N_z/2}^{N_z/2} F(k_i, k_k)\, T(k_i, k_k) \times e^{i(k_i x + k_k z)} \tag{21}$$

to yield the fractal C_L product multipliers f_{CL} at t_1 and t_2. Note that the inverse transform in the y-direction of equation (13) has been omitted, because it appears as a Gaussian multiplier in the final result. The inverse transform is performed first along the z direction, then along x. In the example plume that is embedded within the STATBIC-2D demonstration code, meanderings of the plume center line in the vertical plane due to fractal vertical wind statistics are also included. Application of equations (21) and (14) to equations (16) and (17) then provides a 2D radiance distribution for the plume, in the (x,z) plane.

Prospective application developers of STATBIC-2D should note that they must provide an integrated crosswind optical depth for any plumes that they might adapt to the algorithm. This parameter appears in the program driver as the TAU variable (and as the t quantity of eq (16)). The user thus must supply an obscurant mass extinction coefficient α_{ext} and calculate the C_L product for the mean plume—$(C_L(x,z))_m$ of equation (14)—in order to obtain TAU at each crosswind LOS.

Currently, the STATBIC algorithms are configured as demonstration programs that are not flexible in terms of user input, particularly STATBIC-2D. User input for this demonstration code (detailed below) is limited to an ending time and a random-number seed. The emission rate of the source and transport characteristics of the boundary layer medium are preset and are not user-modifiable in the present version of STATBIC-2D. STATBIC-3D allows the user to have more control over the characteristics of the concentration field, but omits simulation of advected transport of that field through the scenario array.

3.2 Platform-Specific Comments

STATBIC source code and executables are available in packages for either a generic IBM PC with VGA-level graphics (coded in FORTRAN) or a Silicon Graphics, Inc. (SGI), minicomputer using SGI GL (graphics language) graphics (coded in FORTRAN and C++).

3.2.1 PC Version

In the PC edition, only STATBIC-2D presently makes use of video graphics to display a time-stepped "movie" of the notional radiance of a sample steady-state plume emitted from a point source. The plume is viewed in orthographic (parallel) projection. Two simultaneous views of the integrated path radiance of the plume are shown. The top half of the image shows the radiance field with the illumination source behind the plume ("in the observer's eyes"); the bottom half is for the situation when the

sun is behind the observer. Both depictions rely on simple models (described in sect. 2) for the limiting path radiance field within the plume, and integration to obtain an apparent path radiance.

The distribution disk contains both the PC and SGI versions of the STATBIC demonstration code. The root directory contains copies of the PKZIP and PKUNZIP file-compression utilities (version 2.04g) required to install the software, along with a text file describing the diskette contents. The PC version of the package is archived in the PCVER.ZIP file in the /PC subdirectory. PKUNZIP installs the following files for the PC version:

- README.1ST—PC package overview (text file)
- README—Additional implementation comments (text file)
- D3FRCTAL.FOR—STATBIC-3D source code
- D2PLUME.FOR—STATBIC-2D source code
- VGALIBDH.C—C source code for special VGA driver
- COMPSTAT.BAT—Batch file for compilation and linking of PC package
- VGALIBDH.OBJ—Object code for VGA graphics driver to be linked with D2PLUME.OBJ
- VGALIBDH.SAV (same as VGALIBDH.OBJ)
- D3FRCTAL.EXE—STATBIC-3D executable code for PC
- D2PLUME.EXE—STATBIC-2D executable code for PC
- CDEMO002.DAT—Sample binary output concentration file for STATBIC-3D

After installation, the user need only run either D3FRCTAL.EXE or D2PLUME.EXE to obtain immediate results. Any modifications to the source code will require execution of the COMPSTAT.BAT batch file to recompile and link the modified package. The FORTRAN compiler used for this version is Microsoft FORTRAN. Because modification of the custom VGA interface is not recommended, no C compiler should be required for the demonstration package. Also, no data reformatting or graphics postprocessing software is provided with the PC package.

3.2.2 *SGI Version*

The SGI version of STATBIC is first installed onto a PC from the /SGI/SGIVER.ZIP file and is then transferred (via FTP, for example) to the host SGI machine. After transfer to the SGI, the user must set the permission mask on the NEWNAME batch file with the CHMOD utility (using a mask such as "777" or "$a + x$"). NEWNAME is then executed to rename the set of files to the SGI convention, which permits longer file names than are allowed on the PC. The installed files and their significance are the following (PC file names are in parentheses after the SGI names):

- newname (NEWNAME)—Batch file for renaming from PC convention to SGI convention.
- how2use.sgi (HOW2USE.SGI)—SGI package overview (text file).
- how2use.shw (HOW2USE.SHW)—Outline for the use of the showconc routine (text file).
- comp_all (COMP_ALL)—Batch file that compiles and links both the STATBIC-3D and STATBIC-2D demonstration packages, as well as utilities.
- comp_3d (COMP_3D)—Batch file that compiles and links the STATBIC-3D package (d3frctal) and its showconc utility.
- comp_demo (COMP_DMO)—Batch file that compiles and links the STATBIC-2D package and its utilities.
- comp_save (COMP_SAV)—Batch file that compiles and links the STATBIC-2D irisplume.f code.
- comp_flip (COMP_FLP)—Batch file that compiles and links the animation routines for STATBIC-2D image files.
- d3frctal.f (D3FRCTAL.F)—Source code for STATBIC-3D.
- showconc.f (SHOWCONC.F)—Source code for SGI/GL viewer of STATBIC-3D output concentration files.
- irisplume.f (IRISPLUM.F)—Source code for version of STATBIC-2D that does not display to the screen during execution. Output goes to animation file, which may be read by the donsbw display postprocessor.
- rtirisplume.f (RTIRISPL.F)—Source code for STATBIC-2D version similar to irisplume. Top panel shows "sun-in-eyes" case, and bottom panel shows sun-behind-observer case. Results are displayed on screen during run.
- rtirisgaus.f (RTIRISGA.F)—Source code for STATBIC-2D version similar to rtirisplume except that top panel shows fractal plume and bottom panel shows smooth Gaussian plume. "Sun-in-eyes" illumination is used in both cases. This version also generates an RGB output animation file that may be displayed by the anyflip utility.
- rtirisovrhd.f (RTIRISOV.F)—Source code for STATBIC-2D version similar to rtirisplume, except that view is from overhead and illumination is from the side.
- rtirisex1.f (RTIRISEX.F)—Source code for a STATBIC-2D version that is similar to rtirisplume, except that the limiting path radiance map has been modified to show highlighted regions where single scattering is dominant; areas where multiple scattering dominates are dark.
- rtirispuff.f (RTIRISPU.F)—Source code for a STATBIC-2D version that is similar to rtirisplume, except that down-wind advection of a single Gaussian puff emitted at time $t = 0$ s is shown.

- rtgraphic.c (RTGRAPHI.C)—Source C routines for screen display *during* STATBIC-2D RT-series runs.
- anyflip.f (ANYFLIP.F)—Source code for STATBIC-2D post-run animator for RGB-format files. Currently, only rtirisgaus outputs animation files compatible with this utility.
- donsbw.f (DONSBW.F)—Source code for STATBIC-2D animator for gray-scale files output by irisplume. The donsbw input file must be named "flip.dat".
- rtplot.c (RTPLOT.C)—C source code for anyflip and donsbw graphics drivers.
- confr001.dat (CONFR001.DAT)—Sample output concentration file output by the STATBIC-3D code for time t = 1 s (unformatted). This file may be used to demonstrate operation of the showconc utility.
- confr002.dat (CONFR002.DAT)—Sample output concentration file output by the STATBIC-3D code for time t = 2 s (unformatted). This file may be used to demonstrate operation of the showconc utility.

The SGI implementation of both STATBIC-3D and STATBIC-2D thus has a number of useful options and utilities not present in the PC version. SGI systems typically have many powerful image manipulation utilities available to edit output images for publication or in combination with other imagery. It is thus appropriate that the most versatile edition of STATBIC is configured for the SGI platform.

4. User's Guide

Input to the present (demonstration) editions of the STATBIC codes is simple, because these editions rely on preset parameters in the codes. Input to the STATBIC programs is interactive in the present implementation of the algorithm.

4.1 STATBIC-3D

User input to STATBIC-3D consists of six to eight lines of free-format parameters, depending on the choice of fractal or (for test purposes) uniform or smooth concentration output from the code:

(1) NVX, NVY, NVZ

NVX, NVY, and NVZ are the array dimensions in the x, y, and z directions in the scenario array. Each of these dimensions must be equal to an integral power of 2, because of the choice of FFT implemented in the code. The upper bounds on these dimensions are set by the MAXNXS, MAXNYS, and MAXNZS values in a PARAMETER statement at the beginning of the code.

(2) TARGOD

The single Gaussian puff or uniform obscurant distribution considered by STATBIC-3D must be normalized by the CONORM parameter discussed below. As a guide before input of CONORM, the user may specify an optical depth along the y-axis with the TARGOD parameter. The code assumes a nominal mass extinction coefficient ALPHA equal to 4.302 $m^2\ g^{-1}$, which is appropriate to fog oil at visible wavelengths. Also, the code assumes that each cubical cell has an edge length of 1 m. The code then outputs an average concentration TARCON, given by the relation

TARCON = TARGOD / (ALPHA * float(NVY)),

which the user may consider before setting the following CONORM value.

(3) CONORM

The maximum of the mean concentration distribution is given by the CONORM parameter (units of g m^{-3}).

(4) IOTYPE

Selection of the concentration field type produced by the STATBIC-3D code is controlled by the IOTYPE switch. Valid selections for this switch and their meaning are the following:

1 = uniform concentration in the array volume, set equal to the CONORM value given above.

2 = fractal concentration throughout the array volume, with a mean concentration of CONORM.

3 = smooth Gaussian puff concentration distribution centered at middle of array volume. The distribution has the mathematical form

$$C_{ijk}\,\frac{\text{CONORM}}{\text{SNORM}}\exp\left[-\frac{1}{2}\left(\frac{x_i^2}{\sigma_{xb}^2}+\frac{y_j^2}{\sigma_{yb}^2}+\frac{z_k^2}{\sigma_{zb}^2}\right)\right], \tag{22}$$

where C_{ijk} is the concentration (g m^{-3}),

$$x_i = \frac{\text{float}\,(i) - \text{float}\,(\text{NVX})}{2},$$

$$y_j = \frac{\text{float}\,(j) - \text{float}\,(\text{NVY})}{2},$$

$$z_k = \frac{\text{float}\,(k) - \text{float}\,(\text{NVZ})}{2},$$

are the coordinates of the point (i,j,k) relative to the scenario array center (in dimensionless units of cell edge lengths),

$$\sigma_{xb} = \frac{\text{float}\,(\text{NVX})}{4.3},$$

$$\sigma_{yb} = \frac{\text{float}\,(\text{NVY})}{4.3},$$

$$\sigma_{zb} = \frac{\text{float}\,(\text{NVZ})}{4.3},$$

and SNORM = $(2\pi)^{3/2}\,\sigma_{xb}\,\sigma_{yb}\,\sigma_{zb}$.

The sigmas given above are also in dimensionless units of cell-edge lengths (in the code, the sigmas appear as sigxb, sigyb, and sigzb). The factor of 4.3 above is essentially arbitrary; this value tapers the Gaussian distribution to a value of about 10 percent of the peak central value and thus bounds the distribution in a region where it is "significantly" above zero.

4 = Fractal Gaussian puff concentration distribution. This option applies a fractal filter to the smooth distribution described above under the IOTYPE = 3 selection.

(5) USRFIL (for IOTYPE = 1 or 3, uniform or smooth distribution cases)

When a uniform or smooth Gaussian puff distribution is used, the name of the file that receives the output is given by the CHARACTER*80 variable USRFIL. (Note that PC users must use a name that conforms to DOS limitations for file names; i.e., the filename is at most 8 characters long, with an additional "." and three-character extension, for a total of 12 characters.)

(6) SEED (for IOTYPE = 2 or 4, fractal concentration case)

The user supplies a real number SEED to initialize the random number generator, which supplies the random phases required to produce the $T(k_i,k_j,k_k)$ function of equation (19).

(7) ENDTIM (for IOTYPE = 2 or 4, fractal concentration case)

The time at which the evolution of the STATBIC-3D algorithm is terminated (in seconds) is given by the ENDTIM parameter. Results are produced at 1-s intervals, so an integer value of ENDTIM is appropriate. At least two output files are supplied (even if the user gives ENDTIM a value of 0 s), because the multiplexing of the FFT algorithm automatically provides a second event time separated from the first by 1 s.

(8) ROOTNM (for IOTYPE = 2 or 4, fractal concentration case)

Output concentration files produced under the fractal options are stored under names derived from the ROOTNM parameter. The user is prompted to supply a string of five characters in ROOTNM, which forms the prefix for a sequence of filenames for successive time steps. If the name supplied in ROOTNM is less than five characters long, ROOTNM is padded to five characters with trailing "X" characters. The particular time to which the given file pertains is designated by the next three numeric characters after ROOTNM: "001" for the first time, "002" for the second time, etc. The file extension ".DAT" is then appended to complete the output file name. Examples of valid file names for an initial time and the initial time plus 1 s are "CLOUD001.DAT" and "CLOUD002.DAT," where the user has supplied the string "CLOUD" in the ROOTNM variable.

Other parameters are assigned values in STATBIC-3D that may be broken out as explicit user input in future versions of the algorithm. In general, these are used to calculate spatial or temporal correlation factors for Kolmogorov turbulence within the planetary boundary layer or to specify the times at which results are generated. The variable names, assigned values, and meanings for these parameters are the following:

(1) Timing variables:

- TIME = 0.0 (initial time minus DELT1, in seconds)

  (In subsequent assignments, TIME is the scenario time in seconds from the time the cloud evolution starts, i.e., TIME = TIME + DELT1 or TIME = TIME + DELT2.)

- DELT1 = 1.0 (time increment between transform sets, in seconds)

- DELT2 = 1.0 (time increment between the two members of a transform set, in seconds)

  Because the algorithm calculates results for a pair of times with each transform, the option exists to use one time increment between the two members of the transform pair (DELT2) and between the last member of a given transform pair and the initial member of the subsequent transform pair (DELT1). For example, the first two transform pairs yield the following four scenario times:

First transform:

TIME(1) = DELT1
TIME(2) = DELT1 + DELT2

Second transform:

TIME(3) = DELT1 + DELT2 + DELT1
TIME(4) = DELT1 + DELT2 + DELT1 + DELT2

With the assigned values stated above, the scenario time begins at TIME = 1 s and has a uniform spacing of 1 s for all subsequent times. When the value of TIME computed for the last member of a transform pair equals or exceeds the value of ENDTIM, the program terminates *after* the results for this last transform pair are written. With the assigned values above, this implies that results for TIME = 1 s and TIME = 2 s will be recorded, even if an ENDTIM value of 0.0 s is input by the user.

(2) Boundary layer parameters for correlation scaling:

- UBAR = 3.0 (mean wind speed, in meters per second)
- UHGHT = 2.0 (height of mean wind speed, in meters)
- ZNOT = 0.005 (surface roughness height, in meters)

These parameters are then combined with previous user-input or assigned parameters to produce a chain of derived quantities leading to calculation of correlation factors:

- USTAR = 0.4*UBAR/alog(UHGHT/ZNOT) (friction velocity, in meters per second)
- XOUTER = float(NVX) (outer scale on x-axis, in meters)
- YOUTER = float(NVY) (outer scale on y-axis, in meters)
- ZOUTER = float(NVZ) (outer scale on z-axis, in meters)
- TOUTER = sqrt(XOUTER*ZOUTER)/(2.0*USTAR) (outer scale in time, in seconds)
- TSMALL = DELT1/TOUTER (scaled time used as argument of correlation function for the DELT1 interval)
- TLARGE = DELT2/TOUTER (scaled time used as argument of correlation function for the DELT1 interval)

Program output from STATBIC-3D is routed to either the USRFIL file (for the uniform or smooth time-independent obscurant concentration cases) or to the (ROOTNM)NNN.DAT file (for the nonadvected fractal concentration field).

4.2 STATBIC-2D

Input to the STATBIC-2D demonstration program consists only of an initial random number seed and a scenario ending time:

- LSEED

  The user supplies an integer number LSEED to initialize the random number generator, which supplies the random phases required to produce the $T(k_i,k_k)$ function of equation (21).

- ENDTIM

  The time at which the evolution of the STATBIC-2D demonstration program is terminated (in seconds) is given by the ENDTIM parameter. Results are produced at 1-s intervals, so an integer value of ENDTIM is appropriate.

  As mentioned in section 3.2.2 on the SGI implementation of STATBIC-2D, program output may be routed to either the screen or output files. Thus, in some SGI versions of STATBIC-2D, the user is also prompted to enter the name of a file where output images are stored for animation playback by anyflip, donsbw, or other graphics postprocessor.

5. Sample Cases

The STATBIC algorithm is easily applied to example cases that make use of the pre-set parameters in the compiled code. More difficult is the adaptation of the algorithm to user-defined environmental scenarios, obscurant types, and emission rates. It is also left to the user to reformat the run results and tailor them to specific applications, such as simulator displays or statistical analysis in wargames. For the user who wishes to employ the STATBIC algorithm in such contexts, we provide a few simple examples showing the code's capabilities.

5.1 Uniform Fractal Field (PC Implementation)

The PC or SGI version of STATBIC-3D may be used to generate fractal fields that are homogeneous or "uniform" over the scenario array volume. The user makes this selection by setting the IOTYPE parameter described above to a value of 2. The array used here has $x \times y \times z$ dimensions of $64 \times 32 \times 16$ m. The assumed mass extinction coefficient used here is 4.302 $m^2\ g^{-1}$. The TARGOD optical depth parameter is set to 10 in this case, yielding a value of 0.07264 g m^{-3} for the mean concentration. This is also the value that the user enters as the CONORM parameter. Figure 3 shows four adjacent horizontal layer sections of the resulting fractal field at four successive times (with ENDTIM = 4 s), using a 16-level gray scale. The top row depicts the evolving concentration field for layer 7 (out of 16, with 1 as the lowest layer) at times $t = 1$ s (extreme left frame) to $t = 4$ s (extreme right frame). The next three rows down represent results for layers 8, 9, and 10, respectively.

Careful examination of the density patterns within each layer shows fluctuations that strengthen with time and others that weaken. Correlations of fluctuation patterns between layers are also apparent in the results. The patterns displayed in this example are not transported laterally across the scenario array by the wind. Evolution of the density pattern is thus due to the time constant for fluctuation decay and the time correlation statistics given by equations (5) and (6).

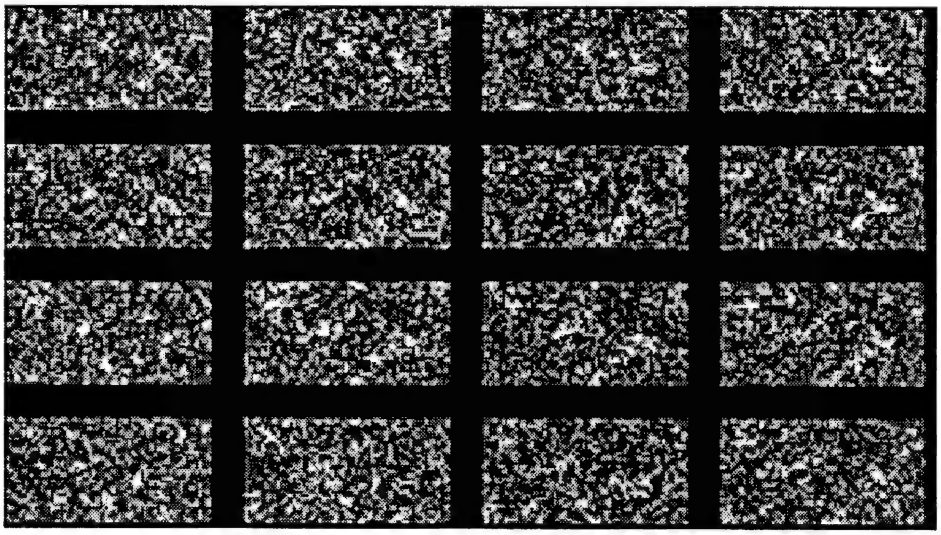

Figure 3. Example case for STATBIC-3D PC implementation; uniform fractal concentration field; rows 1–4 correspond to scenario array horizontal layers 7–10; columns 1–4 correspond to times of 1–4 s.

5.2 Smooth Gaussian-Puff Concentration Field (PC Implementation)

For comparison and checkout purposes, the IOTYPE = 3 option allows the user to create a smooth Gaussian puff distribution of the form given in equation (22). The PC version of STATBIC-3D produces the sectioned distribution shown in figure 4 with this parameter choice. The same $64 \times 32 \times 16$ m scenario array and 16-level gray scale used in the previous example are used for this case. Unlike figure 3, figure 4 shows the horizontal (xy) layers from the bottom ($k = 1$, upper left) to the top ($k = 16$, lower right). The maximum concentration for this example (1.12×10^{-5} g m^{-3}) is given by the ratio CONORM/SNORM of equation (22).

5.3 Fractal Gaussian-Puff Concentration Field (PC Implementation)

The selection IOTYPE = 4 imposes a fractal filter upon the smooth Gaussian-puff distribution shown in figure 4. In this case, the maximum concentration found (1.85×10^{-5} g m^{-3}) is somewhat larger than in the example above. Figure 5 is a display similar in format to figure 3, showing the time-stepped evolution of the four middle layers in the concentra-

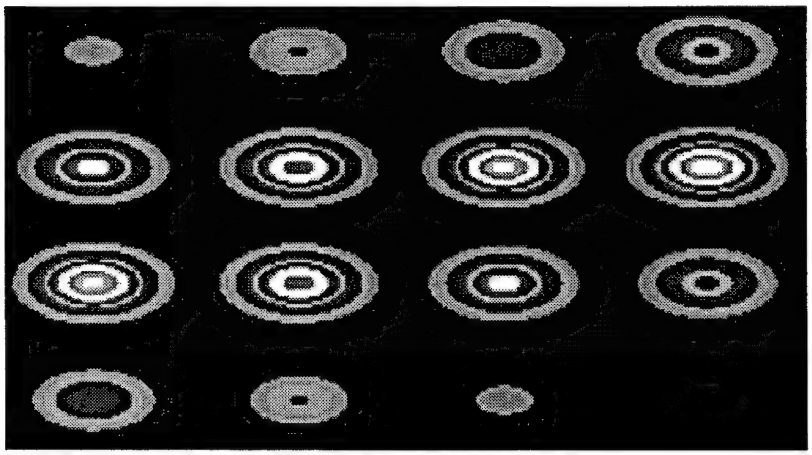

Figure 4. PC version of STATBIC-3D for smooth Gaussian concentration field; horizontal sections from lowest (layer 1, upper left) to highest (layer 16, lower right).

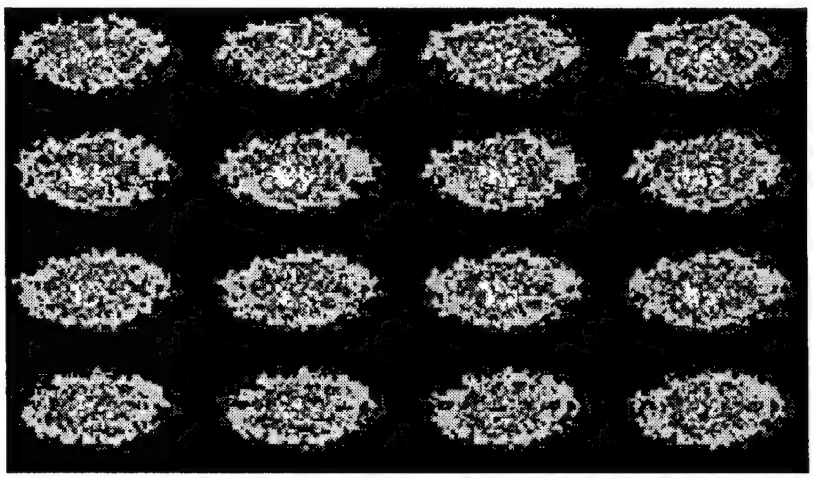

Figure 5. PC STATBIC-3D sample case for fractal Gaussian concentration field; rows 1–4 correspond to horizontal section layers 7–10; columns 1–4 correspond to times of 1–4 s.

tion field. The correlation of fluctuating density structures between adjacent times and layers is again apparent. Note that the demonstration software used here does not allow the puff to be blown across the scenario array. It is also apparent from this example that the puff is not allowed to diffuse out of the scenario array box. Both of these features may be added by the user at a later point.

5.4 Uniform Fractal Field (SGI Implementation)

On an SGI system, the example CONFR001.DAT and CONFR002.DAT binary-data files produced by the d3frctal (STATBIC-3D) program may be examined with the showconc viewer. The screen output of showconc is in color; red, green, and blue colors are used to highlight each of the x,y,z planes in the scenario array. A "gray scale" is used in each color to show the relative concentration from the lowest (black) to the highest (brightest). The user may step through the scenario array with the keyboard function keys: F1 and F2 step backward and forward in the x-direction (red plane), respectively. F3 and F4 move backward and forward in the y-direction (green plane); F5 and F6 perform the same respective functions in the z-direction (blue plane). The user may also embed the highlighted planes in a wire-frame that shows the position of all cell boundaries in the scenario array. The wire-frame may be turned on and off with the F11 and F12 keys, respectively (at program entry, the wire-frame is present). The user may rotate the entire scenario array with the system mouse (the default orientation at program entry is restored with the "escape" key). Zooming of the entire array is performed with the "page up" and "page down" keys.

5.5 Fractal and Smooth Gaussian Plumes (SGI Implementation)

The fractal 2D plume may be shown above its parent Gaussian plume with the rtirisgaus (RTIRISGA) SGI implementation of STATBIC-2D. This version of the STATBIC algorithm produces both an on-screen view of the comparison and a "flip-file" that may be viewed later with the anyflip animator. Images are stored at 1-s time increments in the animation file. When running anyflip, the user enters a real number between 0.0 (slowest) and 1.0 (fastest) to control the speed of the animation. A negative value terminates the program. Figure 6 shows a typical sample frame from rtirisgaus. Both views shown here are for the "sun-in-the-eyes" forward-scattering geometry.

5.6 Fractal Plume, Varying Sun Angle (SGI Implementation)

The irisplume (IRISPLUM) and rtirisplume (RTIRISPL) STATBIC-2D SGI programs may be used to demonstrate fractal plume radiance as a function of illumination direction. As mentioned previously, a radiative transport code was used to determine the average limiting-path radiance

over the visible face of a uniform cubic fog oil cloud for forward scattering ("sun-in-eyes") and backward scattering ("sun-behind-back"), as a function of axial optical depth (eq (15)). The solar elevation was set at 30° in both cases. Figure 7 displays screen output from the rtirisplume program with the forward-scattering case above the backscattering case. The obscurant source emission rate is held constant in this example. The plume was allowed to meander in the vertical by fractal variation of the vertical wind component. Note the reversal of cloud core radiance relative to the radiance at the outside edge of the plume between the two geometries. Also note, in the forward-scattering view, that the large hole in the plume has a dense knot to its upwind side.

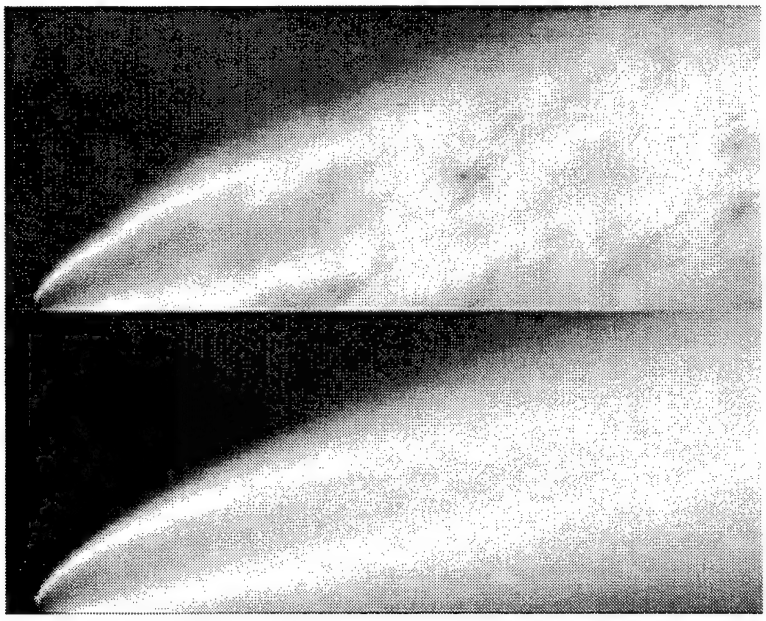

Figure 6. Comparison of STATBIC-2D fractal (top) and smooth Gaussian (bottom) cases.

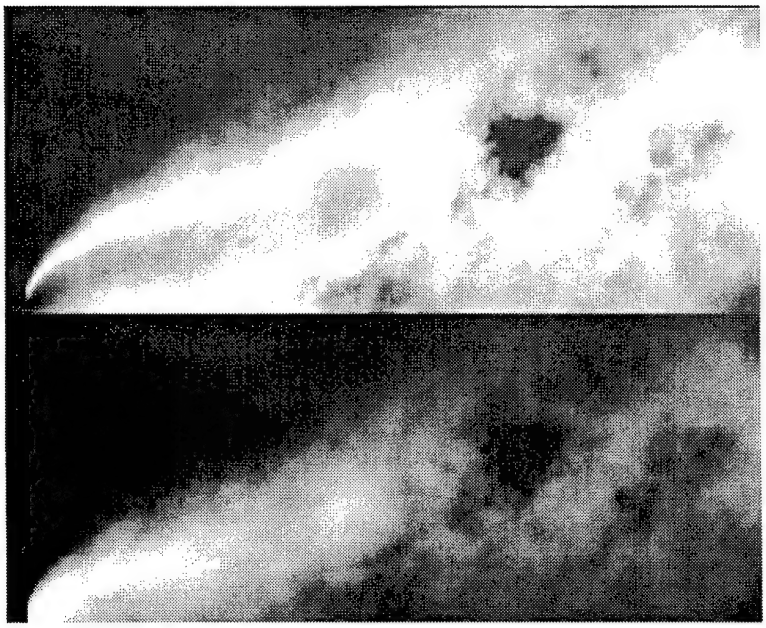

Figure 7. STATBIC-2D comparison for forward-scattered radiance (top) and backscattered radiance (bottom) for fog oil cloud.

5.7 Fractal Puff, Varying Sun Angle, and Time (SGI Implementation)

The STATBIC algorithm for the 2D case may be adapted to cases where a single Gaussian puff is transported across the scenario array and is also allowed to expand as it moves. The modification (implemented as the rtirispuff (RTIRISPU) program) makes use of the 2D description for the horizontal wind speed, plume meander, and diffusive spread for the Gaussian plume case.

6. Conclusions

It should be emphasized that STATBIC is an *algorithm*, not an operational model package. The code that does exist is for a specific illustration of the algorithm's potential for application to obscurant screening problems. Practical real-time usage of the algorithm is, at present, limited to more powerful computers than PCs or low-end SGI machines. Execution of STATBIC for typical sample problems takes several seconds per time step for this latter class of machines, even with the doubled FFT packing strategy.

As currently distributed, the STATBIC demonstration package is written primarily in FORTRAN, with a few support routines written in C. A version of STATBIC-2D has been translated to the C language (by D. Tofsted of ARL); this version is available to DoD users from the authors.

References

1. D. W. Hoock, S. G. O'Brien, J. C. Giever, S. J. McGee, and M. A. Torres, *Physics-Based Environmental and Embedded Process Models for Virtual Simulations*, Proceedings of the 17th ITSEC Conference, Albuquerque, New Mexico, (November 1995).

2. D. W. Hoock, *Statistical Texturing Application to Battlefield Induced Clouds (STATBIC)*, MORS Military Operations Research Handbook, Volume 1: Terrain, Unit Movement, and Environment, edited by W. K. Olson, Military Operations Research Society, pp 4–8, 4–15 (April 1994).

3. D. W. Hoock, *Theoretical and Measured Fractal Dimensions for Battlefield Aerosol Cloud Visualization and Transmission*, Proceedings of the Battlefield Atmospheric Conference, Fort Bliss, Texas, pp 46–55 (December 1991).

4. D. W. Hoock, P. S. Hansen, J. C. Giever, and S. G. O'Brien, *Visualization of Obscuration and Contrast Effects Using the BEAMS Models*, Proceedings of the Battlefield Atmospherics Conference, Las Cruces, New Mexico, pp 3–13 (December 1994).

5. W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes: The Art of Scientific Computing (FORTRAN Version)*, Cambridge University Press, p 397 (New York, 1989).

Distribution

Admnstr
Defns Techl Info Ctr
Attn DTIC-OCP
8725 John J Kingman Rd Ste 0944
FT Belvoir VA 22060-6218

Ofc of the Dir Rsrch and Engrg
Attn R Menz
Pentagon Rm 3E1089
Washington DC 20301-3080

Ofc of the Secy of Defns
Attn ODDRE (R&AT) G Singley
Attn ODDRE (R&AT) S Gontarek
The Pentagon
Washington DC 20301-3080

OSD
Attn OUSD(A&T)/ODDDR&E(R) R Tru
Washington DC 20301-7100

ARL Chemical Biology Nuc Effects Div
Attn AMSRL-SL-CO
Aberdeen Proving Ground MD 21005-5423

Army Communications Elec Ctr for EW RSTA
Attn AMSEL-EW-D
FT Monmouth NJ 07703-5303

Army Corps of Engrs
Engr Topographics Lab
Attn ETL-GS-LB
FT Belvoir VA 22060

Army Field Artillery School
Attn ATSF-TSM-TA
FT Sill OK 73503-5000

Army Foreign Sci Tech Ctr
Attn CM
220 7th Stret NE
Charlottesville VA 22901-5396

Army Infantry
Attn ATSH-CD-CS-OR E Dutoit
FT Benning GA 30905-5090

Army Rsrch Ofc
Attn AMXRO-GS Bach
PO Box 12211
Research Triangle Park NC 27709

Army Strat Defns Cmnd
Attn CSSD-SL-L Lilly
PO Box 1500
Huntsville AL 35807-3801

CECOM
Attn PM GPS COL S Young
FT Monmouth NJ 07703

CECOM RDEC Elect System Div Dir
Attn J Niemela
FT Monmouth NJ 07703

CECOM
Sp & Terrestrial Commctn Div
Attn AMSEL-RD-ST-MC-M H Soicher
FT Monmouth NJ 07703-5203

Dpty Assist Secy for Rsrch & Techl
Attn SARD-TT F Milton Rm 3E479
The Pentagon
Washington DC 20301-0103

Hdqtrs Dept of the Army
Attn DAMO-FDT D Schmidt
400 Army Pentagon Rm 3C514
Washington DC 20301-0460

MICOM RDEC
Attn AMSMI-RD W C McCorkle
Redstone Arsenal AL 35898-5240

Natl Security Agency
Attn W21 Longbothum
9800 Savage Rd
FT George G Meade MD 20755-6000

Science & Technology
101 Research Dr
Hampton VA 23666-1340

US Army CECOM Rsrch, Dev, & Engrg
Attn R F Giordano
FT Monmouth NJ 07703-5201

US Army Edgewood Rsrch, Dev, & Engrg Ctr
Attn SCBRD-TD J Vervier
Aberdeen Proving Ground MD 21010-5423

Distribution

US Army Info Sys Engrg Cmnd
Attn ASQB-OTD F Jenia
FT Huachuca AZ 85613-5300

US Army Materiel Sys Analysis Activity
Attn AMXSY-CR Marchetti
Attn AMXSY-D J McCarthy
Aberdeen Proving Ground MD 21005-5071

US Army Matl Cmnd
Dpty CG for RDE Hdqtrs
Attn AMCRD BG Beauchamp
5001 Eisenhower Ave
Alexandria VA 22333-0001

US Army Matl Cmnd
Prin Dpty for Acquisition Hdqrts
Attn AMCDCG-A D Adams
5001 Eisenhower Ave
Alexandria VA 22333-0001

US Army Matl Cmnd
Prin Dpty for Techlgy Hdqrts
Attn AMCDCG-T M Fisette
5001 Eisenhower Ave
Alexandria VA 22333-0001

US Army Mis Cmnd (USAMICOM)
Attn AMSMI-RD-CS-R Documents
Redstone Arsenal AL 35898-5400

US Army Natick Rsrch, Dev, & Engrg Ctr
Acting Techl Dir
Attn SSCNC-T P Brandler
Natick MA 01760-5002

US Army Rsrch Ofc
Attn G Iafrate
4300 S Miami Blvd
Research Triangle Park NC 27709

US Army Simulation, Train, & Instrmntn Cmnd
Attn J Stahl
12350 Research Parkway
Orlando FL 32826-3726

US Army Tank-Automtv & Armaments Cmnd
Attn AMSTA-AR-TD C Spinelli
Bldg 1
Picatinny Arsenal NJ 07806-5000

US Army Tank-Automtv Cmnd Rsrch, Dev, & Engrg Ctr
Attn AMSTA-TA J Chapin
Warren MI 48397-5000

US Army Test & Eval Cmnd
Attn R G Pollard III
Aberdeen Proving Ground MD 21005-5055

US Army Topo Engrg Ctr
Attn CETEC-ZC
FT Belvoir VA 22060-5546

US Army Train & Doctrine Cmnd
Battle Lab Integration & Techl Dirctrt
Attn ATCD-B J A Klevecz
FT Monroe VA 23651-5850

US Military Academy
Dept of Mathematical Sci
Attn MAJ D Engen
West Point NY 10996

USAASA
Attn MOAS-AI W Parron
9325 Gunston Rd Ste N319
FT Belvoir VA 22060-5582

USArmy TRADOC
Attn ATRC-WEC D Dixon
White Sands Missile Range NM 88002-5501

USATRADOC
Attn ATCD-FA
FT Monroe VA 23651-5170

Nav Air War Cen Wpn Div
Attn CMD 420000D C0245 A Shlanta
1 Admin Cir
China Lake CA 93555-6001

Nav Surface Warfare Ctr
Attn Code B07 J Pennella
17320 Dahlgren Rd Bldg 1470 Rm 1101
Dahlgren VA 22448-5100

Naval Surface Weapons Ctr
Attn Code G63
Dahlgren VA 22448-5000

Distribution

AFSPC/DRFN
Attn CAPT R Koon
150 Vandenberg Stret Ste 1105
Peterson AFB CO 80914-45900

Air Weather Service
Attn TechL Lib FL4414 3
Scott AFB IL 62225-5458

ASC OL/YUH
Attn JDAM-PIP LT V Jolley
102 W D Ave
Eglin AFB FL 32542

GPS Joint Prog Ofc Dir
Attn COL J Clay
2435 Vela Way Ste 1613
Los Angeles AFB CA 90245-5500

Phillips Lab
Atmospheric Sci Div Geophysics Dirctrt
Hanscom AFB MA 01731-5000

Phillips Lab
Attn PL/LYP
Attn PL/LYP Chisholm
Hanscom AFB MA 01731-5000

USAF Rome Lab Tech
Attn Corridor W Ste 262 RL SUL
26 Electr Pkwy Bldg 106
Griffiss AFB NY 13441-4514

DARPA
Attn B Kaspar
Attn L Stotts
3701 N Fairfax Dr
Arlington VA 22203-1714

NASA Spct Flt Ctr Atmospheric Sciences Div
Attn Code ED 41 1
Huntsville AL 35812

NIST
Attn MS 847.5 M Weiss
325 Broadway
Boulder CO 80303

ARL Electromag Group
Attn Campus Mail Code F0250 A Tucker
University of Texas
Austin TX 78712

Ashtech Inc
Attn S Gourevitch
1177 Kifer Rd
Sunnyvale CA 94086

Dir for MANPRINT
Ofc of the Deputy Chief of Staff for Prsnnl
Attn J Hiller
The Pentagon Rm 2C733
Washington DC 20301-0300

Natl Ctr for Atmospheric Research
Attn NCAR Library Serials
PO Box 3000
Boulder CO 80307-3000

New Mexico State University
Physical Sci Lab
Attn S O'Brien (3 copies)
PO Box 30002
Las Cruces NM 88003-8002

US Army Rsrch Lab
Attn AMSRL-IS-E D Hoock (3 copies)
White Sands Missile Range NM 88002-5001

US Army Rsrch Lab
Attn AMSRL-BE-E
White Sands Missile Range NM 88002-5501

US Army Rsrch Lab
Attn AMSRL-CI-LL Techl Lib (3 copies)
Attn AMSRL-CS-AL-TA Mail & Records Mgmt
Attn AMSRL-CS-AL-TP Techl Pub (3 copies)
Attn AMSRL-IS J Gantt
Attn AMSRL-IS-E K Gurton (20 copies)
Adelphi MD 20783-1197

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	April 1998	Final, from October to November 1996

4. TITLE AND SUBTITLE
STATBIC—A Method for Inclusion of Fractal Statistics in Obscurant Transport Models

5. FUNDING NUMBERS
DA PR: B53A
PE: 61102A

6. AUTHOR(S)
Sean G. O'Brien (Physical Science Laboratory, New Mexico State University) and Donald W. Hoock (Army Research Laboatory)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
Attn: AMSRL-IS-E (dhoock@arl.mil)
White Sands Missile Range, MM 88002-5501

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-1375

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
2800 Powder Mill Road
Adelphi, MD 20783-1197

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
AMS code: 61110253A11
ARL PR: 7FEJ70

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

A generic texture routine was developed for upgrading smooth obscurant cloud models by introduction of time- and space-dependent fluctuations in line-of-sight (LOS) propagation and image generation. The routine runs separately from or in conjunction with other obscuration models that predict electro-optical (EO) propagation for mean or average aerosol concentration contributions in the obscurant cloud.

Atmospheric turbulence and eddy structures are the underlying sources of concentration fluctuations. Eddy structures can possess certain well-defined and statistically predictable correlations between larger and smaller eddies that are linked to the steady breakup of larger eddies and the cascade of energy to smaller eddy scales. The resulting effects of these correlations in concentration are simulated by prototype algorithms that provide either two-dimensional propagation overlays for image modification or three-dimensional volume fluctuations. Path-integrated concentration, LOS propagation fluctuations, and realistic cloud imaging are then simulated by multiplication of pseudo-random fluctuation outputs with smooth, ensemble-averaged outputs from more traditional, simple obscuration models.

(cont'd on reverse)

14. SUBJECT TERMS	15. NUMBER OF PAGES
plume, simulation	36
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

13. Abstract (cont'd)

Two platform-specific implementations of the generic texture routine were developed for IBM PC and Silicon Graphics, Inc. (SGI), display hardware. Examples of graphical output for both the two and three-dimensional algorithms are shown.